МАШИНОСТРОЕНИЕ
MACHINE BUILDING



УДК 621.86/.87
https://doi.org/10.23947/2541-9129-2024-8-2-68-77

Научная статья



EDN: BMJTNE

Сравнительный анализ качества работы искусственных нейронных сетей для оценки технического состояния стального каната

Р.В. Хван 
Донской государственный технический университет, г. Ростов-на-Дону, Российская Федерация
✉ khvanroman@yandex.ru

Аннотация
Введение. В настоящее время искусственные нейронные сети (ИНС) успешно применяются для технического диагностирования стальных канатов. Зачастую при этом используют дорогостоящие программные продукты с адаптированной средой реализации нейронных сетей, такие как STATISTICA, Amygdala, MATLAB Simulink. Наиболее доступным способом построения и обучения ИНС с финансовой точки зрения является написание собственного программного кода с использованием интерактивных библиотек, таких как TensorFlow, PyTorch, Scikit-learn. Однако такие библиотеки не являются полноценными адаптированными средами построения ИНС, и для их использования необходимо владеть первичными навыками программирования. Поэтому качество ИНС зависит не только от архитектуры, объема и состава обучающих выборок, но и от метода (среды) построения ИНС. Целью данного исследования является сравнение качества работы ИНС, построенных и обученных различными методами, по критерию тестовой производительности сети, доверительным уровням оценки технического состояния каната, а также трудоемкости и скорости обучения. В связи с этим создано новое программное обеспечение для решения задачи оценки технического состояния стального каната по комбинации различных браковочных показателей.
Материалы и методы. Основой для обучения ИНС послужили статистическая база данных типовых повреждений стальных канатов, экспертная оценка их технического состояния. Программное обеспечение написано на языке программирования Python. Приведены различные методы программирования нейронной сети: ИНС, построенной на базе программного комплекса STATISTICA, и ИНС, построенной с использованием интерактивной библиотеки Scikit-learn. Для проверки работы ИНС было подготовлено 10 тестовых выборок. Оценка качества работы ИНС проводилась по тестовой производительности сети и доверительным вероятностям (уровням активации «победившего» нейрона) определения технического состояния каната.
Результаты исследования. Построение ИНС с использованием интерактивной библиотеки Scikit-learn показало сравнительно большую трудоемкость построения и сравнительно небольшую скорость обучения. Тестовая производительность сети при объеме тестовой выборки 10 оказалась одинаковой для обеих построенных ИНС. При этом обнаружилась разница в показателе среднего доверительного уровня определения технического состояния стального каната по результатам работы ИНС, построенной на базе программного комплекса STATISTICA, и ИНС, построенной с использованием интерактивной библиотеки Scikit-learn.
Обсуждение и заключение. Полученные результаты показали, что ИНС, построенная с использованием программного комплекса STATISTICA, при одинаковой архитектуре и параметрах обучения сети имеет более оптимальные программные алгоритмы по критериям доверительной вероятности и скорости обучения сети по сравнению с ИНС, построенной с использованием бесплатной библиотеки Skicit-learn. Однако показатель тестовой производительности ИНС оказался одинаковым для обеих ИНС. Такой результат обосновывает использование ведущими мировыми научно-исследовательскими и коммерческими центрами в области искусственного интеллекта библиотек TensorFlow, PyTorch, Scikit-learn. Полученный научный результат позволит численно оценить и сравнить качество искусственных нейронных сетей, имеющих одинаковые

https:// bps-journal.ru

© Хван Р.В., 2024

архитектуру и параметры обучения, но построенных различными методами, он будет полезным как для будущих научных исследований в этой области, так и для выбора оптимальной среды построения ИНС в промышленной сфере деятельности.

Ключевые слова: стальной канат, искусственные нейронные сети, оценка технического состояния, Python, Scikit-learn, STATISTICA, браковочные показатели

Благодарности. Автор выражает благодарность коллективу ООО «ИКЦ «Мысль» НГТУ г. Новочеркасска за предоставленную возможность использовать данные эксплуатации стальных канатов, а именно статистическую базу данных типовых повреждений стальных канатов.

Для цитирования. Хван Р.В. Сравнительный анализ качества работы искусственных нейронных сетей для оценки технического состояния стального каната. *Безопасность техногенных и природных систем*. 2024;8(2):68–77. https://doi.org/10.23947/2541-9129-2024-8-2-68-77

Research Article

Comparative Analysis of the Performance of Artificial Neural Networks in Assessing the Technical Condition of Steel Ropes

Roman V. Khvan

Don State Technical University, Rostov-on-Don, Russian Federation

✉ khvanroman@yandex.ru

Abstract

Introduction. Currently, artificial neural networks (ANN) are successfully used for technical diagnostics of steel ropes. Expensive software products with an adapted neural network implementation environment, such as STATISTICA, Amygdala, MatLab Simulink, are often used for this purpose. The most affordable way to build and train an ANN, from a financial point of view, is to write your own program code using interactive libraries such as TensorFlow, PyTorch, Scikit-learn. However, such libraries are not fully adapted for building an ANN, and to use them you need to have basic programming skills. As a result, the quality of an ANN depends not only on its architecture, training data, and composition, but also on the environment in which it is built. The aim of the work was to compare the quality of the ANN, built and trained by various methods according to the criterion of test network performance, confidence levels for assessing the technical condition of the rope, as well as the complexity and speed of training. For this purpose, new software has been developed to solve the problem of assessing the technical condition of a steel rope using a combination of various rejection indicators.

Materials and Methods. The basis for an ANN training was a statistical database of typical damages of steel ropes and, an expert assessment of the technical condition of steel ropes. The software was written in the Python programming language. Various methods of programming a neural network were presented: an ANN built on the basis of the STATISTICA software package and an ANN built using the interactive Scikit-learn library. Ten test samples were prepared to verify the operation of the ANN. The ANN quality was assessed based on the test network performance and confidence probabilities (activation levels of the “winning” neuron) of determining the technical condition of the rope.

Results. The construction of the ANN using the interactive library Scikit-learn showed a relatively high complexity of construction and a relatively low learning rate of the ANN. Test performance of the network, with a test sample size of ten, turned out to be the same for both built ANNs. At the same time, there was a difference in the indicator of the average confidence level for determining the technical condition of a steel rope between the results of the ANN built on the basis of the STATISTICA software package and the ANN built using the Scikit-learn interactive library.

Discussion and Conclusion. The results showed that the ANN built using the STATISTICA software package with the same architecture and network learning parameters had more optimal software algorithms according to the criteria of confidence probability and network learning speed in comparison with the ANN built using the free Skicit-learn library. However, the indicator of the ANN test performance turned out to be the same for both ANNs. This result justified the use of TensorFlow, PyTorch, and Skicit-learn libraries by the world's leading research and commercial centers in the field of artificial intelligence. The obtained scientific result allows us to numerically evaluate and compare the quality of an ANN having the same architecture and learning parameters, but built using different methods. This will be useful for future scientific research in the field and for selecting the optimal environment for constructing ANNs in industrial applications.

Keywords: steel rope, artificial neural networks, technical condition assessment, Python, Skicit-learn, STATISTICA, rejection indicators

Acknowledgements. The author would like to expresses their gratitude to the team at the Engineering and Consulting Center "Mysl' LLC" of the Novocherkassk NSTU for the opportunity to use the data on the operation of steel ropes, specifically the statistical database on typical damages to steel ropes.

For Citation. Khvan RV. Comparative Analysis of the Performance of Artificial Neural Networks in Assessing the Technical Condition of Steel Ropes. *Safety of Technogenic and Natural Systems.* 2024;8(2):68–77. https://doi.org/10.23947/2541-9129-2024-8-2-68-77

Введение. В настоящее время методы оценки технического состояния объектов машиностроения с использованием ИНС получают все большее распространение. Это связанно прежде всего с относительной простотой выявления зависимостей выходных данных (в данном случае это техническое состояние стального каната) от входных данных (различные комбинации браковочных показателей стального каната). Так, для установления корреляционной зависимости технического состояния стального каната от различных комбинаций десяти браковочных показателей, выраженных в процентах от допустимой степени повреждения, требуются и большое количество аналитических операций, и решение многофакторной задачи регрессии. Программное обеспечение, ядром которого является нейронная сеть, облегчит работу и поможет экспертам и начинающим специалистам при принятии решения о возможности дальнейшей эксплуатации стального каната по различным комбинациям браковочных показателей.

Проблемами надежности и безопасности технических систем с применением современных средств искусственного интеллекта занимаются ученые различных отраслей знания. Примерами могут послужить работы В.А. Воронцова, Е.А. Федорова, А.А. Короткого, А.В. Панфилова, Н.Н. Николаева, А.Р. Юсупова, С.В. Жернакова, Т.И. Горевой, Н.Н. Портягина, Г.А. Пюкке, Б.Ч. Месхи, А.Н. Бескопыльного, С.А. Стельмах, И.Ф. Развеевой и др. [1–8]. В работах этих авторов успешно используются технологии нейросетевого моделирования для достижения различных научно-исследовательских и производственных целей, таких как оценка технического состояния авиационных двигателей, бортовых систем космического аппарата, обнаружение и классификация дефектов стального каната. При этом применяются различные архитектуры, параметры обучения ИНС, программные комплексы и среды разработки искусственной нейронной сети, такие как Alyuda NeuroIntelligence, STATISTICA, Amygdala, MatLab Simulink. Также широко практикуется написание собственного программного кода с использованием бесплатных (с открытым исходных кодом) интерактивных библиотек, таких как TensorFlow, PyTorch, Scikit-learn. Однако стоит отметить отсутствие во всех указанных работах сравнения качества работы ИНС, имеющих одинаковую архитектуру, параметры обучения, объем и состав обучающих выборок, но построенных различными методами, т.е. в различных программных средах разработки.

Зачастую использование ИНС в промышленном масштабе связано с использованием дорогостоящих специализированных программных комплексов, обладающих адаптированной средой реализации нейронных сетей и не требующих от пользователя владения навыками программирования. Более доступным способом построения и обучения нейронной сети с финансовой точки зрения является написание собственного программного кода с использованием бесплатных интерактивных библиотек с открытым исходным кодом. Однако данный метод требует от пользователя владения первичными навыками программирования. Библиотека Scikit-learn имеет открытый исходный код, тогда как годовая лицензия программы STATISTICA составляет около 25 тысяч долларов США.

Цель данной работы — провести сравнительный анализ качества работы ИНС для оценки технического состояния стальных канатов по комбинации браковочных показателей, построенных и обученных различными методами, по критериям тестовой производительности сети, трудоемкости и скорости обучения сети, среднему уровню активации «победивших» нейронов сети и с учетом финансовых затрат на реализацию и использование построенных искусственных нейронных сетей. В связи с этим поставлена задача создать новое программное обеспечение для оценки технического состояния стальных канатов по комбинации браковочных показателей с применением ИНС двумя различными способами: ИНС, построенной на базе программного комплекса STATISTICA, и ИНС, построенной с использованием интерактивной библиотеки Scikit-learn.

Материалы и методы. Основой для обучения ИНС послужил опыт эксплуатации стальных канатов, а именно статистическая база данных типовых повреждений стальных канатов, экспертная оценка технического состояния стальных канатов [9–12]. На рис. 1 представлена схема нейронной сети для оценки технического состояния стального каната, используемая во всех ниже представленных методах.

https:// bps-journal.ru

Рис. 1. Схема ИНС для оценки технического состояния стального каната

Ниже представлены параметры, используемые в обоих методах построения ИНС:

1. Заданы так называемые гиперпараметры — параметры, которые не изменяются в процессе обучения сети, это:

– 10 нейронов входного слоя, равные количеству браковочных показателей стального каната (x_1–x_{10});

– количество выходных параметров (нейронов), три возможных состояния стального каната:

1 — работоспособное, эксплуатация разрешена (y_1);

2 — дефекты в допустимых пределах, эксплуатация разрешена с ограничениями (y_2);

3 — достигнуто предельное состояние, эксплуатация запрещена (y_3);

– количество промежуточных слоев, а также количество нейронов в каждом промежуточном слое.

Таким образом, определена архитектура нейронной сети [13].

2. Заданы обучаемые параметры — параметры, которые изменяются (оптимизируются) в процессе обучения сети: значения синаптических весов *w* (сила связи между нейронами) и смещений *b*. Сначала эти значения задаются случайным образом, а далее в процессе обучения нейронной сети они будут оптимально настроены.

3. Реализован алгоритм прямого распространения (Forward propagation), т. е. вычисление нейронной сети на случайных параметрах *w, b*. На данном этапе необходимы обучающие данные, т. е. выборки входных параметров с известными выходными параметрами, как представлено в таблице 1.

Таблица 1

Данные для обучения нейронной сети

№ п/п	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	y
1	75	0	18	0	0	0	44	0	0	0	3
2	21	0	34	0	0	18	0	0	0	0	1
3	0	89	0	66	0	0	0	0	74	0	3
4	78	0	12	0	0	0	0	0	0	0	2
5	55	90	0	0	0	10	0	0	7	0	3
6	0	43	24	0	0	0	0	0	0	34	1
7	0	65	90	0	57	13	0	100	81	0	3
8	19	0	0	0	13	3	0	0	0	0	1
9	0	56	0	30	0	0	0	0	0	0	1
10	0	61	0	0	0	0	0	83	0	0	2
……	0	0	71	0	87	0	0	41	0	0	2
300	0	0	28	14	0	0	28	100	0	0	3

В таблице 1 десять входных параметров (x_1–x_{10}) являются браковочными показателями стальных канатов, выраженными в процентах от предельных значений. Таким образом, учитывается комбинация различных дефектов, которые могут быть выявлены экспертным путем с использованием 3d модели напряженно-деформированного состояния каната. Каждой такой комбинации соответствует определенное техническое состояние каната. При этом объем обучающей выборки 300. Объем контрольной выборки составил 30 выборок. Для проверки работы ИНС подготовлено 10 тестовых выборок, не участвовавших в процессе обучения.

4. Определена функция активации F. Функция активации необходима для добавления нелинейности и некого порогового значения активации на выходе каждого нейрона. Выбираем функцию активации Relu. Ниже представлена формула для определения уровня активации нейрона [14, 15]:

$$y = F\left(x_1 w_1 + x_2 w_2 + x_3 w_3 + b\right),$$

где y — уровень активации нейрона; $x_1 - x_3$ — уровни активации нейронов предыдущего слоя; $w_1 - w_3$ — синоптические веса; b — смещение функции.

На рис. 2 представлена схема формирования уровня активации одного нейрона.

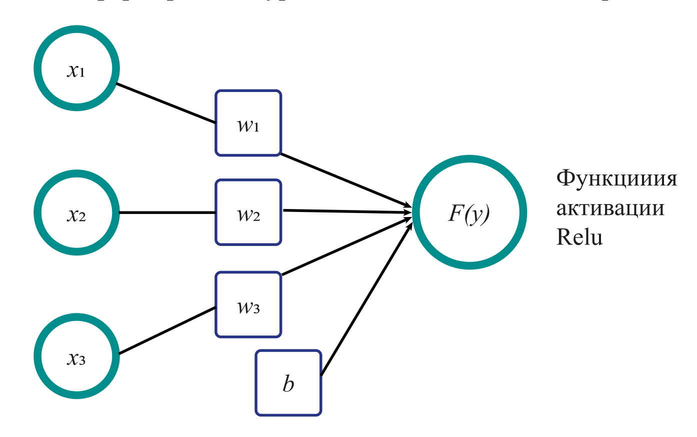


Рис. 2. Схема формирования уровня активации одного нейрона

5. Произведено нормирующее преобразование значений активации нейронов последнего результирующего слоя сети с помощью функции softmax, чтобы их значения были от 0 до 1, и сумма этих значений была равна 1. Другими словами, проведена интерпретация уровней активации нейронов в вероятностном смысле. В результате получено предсказание нейронной сети — состояние стального каната с некоторым доверительным уровнем для текущих значений синаптических весов (связей между нейронами) w и смещений b.

6. Вычисление ошибки E между вычисленными уровнями активации нейронов выходного слоя $y_{выч}$ и целевыми значениями активации нейронов выходного слоя $y_{цел}$ с помощью функции MSE (Евклидово расстояние) или функции кросс-энтропии (используется для определения расстояния между вероятностными распределениями). Следует помнить, что нейронная сеть обучается «с учителем», т. е. на примерах с заранее известными входами и выходами (обучающая выборка).

7. Реализация алгоритма обратного распространения ошибки (backpropagation), целью которого является решение задачи минимизации функции ошибки E в зависимости от синаптических весов (связей между нейронами) w и от смещений b. Для этого использован метод градиентного спуска.

8. Повторение всего алгоритма обучения на следующей обучающей выборке (или группе выборок) с целью минимизации ошибки нейронной сети посредством обновления связей (весов) между нейронами. Каждое такое повторение называют эпохой обучения. Задаем 2000 эпох обучения. Скорость обучения 0,0001. Функция ошибки Adam. После выполнения заданного количества эпох обучения нейронная сеть со всеми настройками сохраняется и может использоваться для предсказания выходных параметров (состояние стального каната) по входным параметрам (комбинации браковочных показателей), ранее не использованным в процессе обучения сети.

STATISTICA содержит два встроенных языка программирования: STATISTICA BASIC и SCL (командный язык). Метод построения нейронной сети в STATISTICA начинается с запуска модуля «Нейронные сети» во вкладке «Добыча данных» на основной рабочей панели программы. Интерфейс программы STATISTICA практически не отличается от известного интерфейса программы MS Office. Для построения ИНС в STATISTICA задаем архитектуру сети и параметры обучения, указанные в выше представленном алгоритме, нажатием соответствующих клавиш с помощью адаптированного интерфейса программы.

Непосредственное обучение нейронной сети активизируется нажатием клавиши «Обучить» в рабочем окне STATISTICA, после чего ИНС проверяется на заранее определенных контрольных выборках. Объем контрольной выборки равен 30. Условие остановки обучения — достижение контрольной производительности сети не ниже 95 %. При этом процесс обучения нейронной сети занял около 20 секунд.

Приведен пошаговый алгоритм для написания программного кода нейронной сети с использованием интерактивной библиотеки Scikit-learn на языке программирования Python [4–7]:

1. В первом блоке устанавливаются дополнительные интерактивные библиотеки (под библиотекой в данном случае понимается совокупность готовых подпрограмм, облегчающих процесс программирования) через ввод следующего кода (в скобках даны пояснения содержания или выполняемой команды кода):

!python -m pip install pandas (библиотека обработки и анализа данных);

!python -m pip install sklearn (библиотека машинного обучения);

!python -m pip install openpyxl (библиотека для работы с файлами Excel).

2. Во втором блоке импортируются модули через ввод следующего кода:

import pandas as pd (модуль обработки и анализа данных);

from sklearn.neural_network import MLPClassifier (модуль для работы с нейронными сетями);

from sklearn.metrics import confusion_matrix, classification_report (дополнительный модуль для работы с нейронными сетями);

import pickle (модуль, использованный для сохранения обученной нейронной сети);

import joblib (модуль, использованный для сохранения обученной нейронной сети).

3. Загрузка базы данных, т. е. набора обучающих данных. Необходимо заранее подготовить файл Excel с обучающими данными, выстроенными в формате, представленном в таблице 1 (рекомендуется более упрощенный вид — только столбцы входных и выходных параметров с их заголовками). Для этого вводим следующий программный код:

ds = pd.read_excel('book1.xlsx'), где book1.xlsx — файл Excel, который должен располагаться в одной папке с файлом создаваемой программы;

ds.head(10) (код для наглядного отображения первых 10 строк таблицы).

4. Далее присваиваются значения переменным X и Y написанием следующего программного кода:

X = ds.drop('Состояние каната',axis=1) (для переменной X присваиваются все столбцы, кроме столбца «Состояние каната» (указывается дословное название столбца);

y = ds['Состояние каната'] (для переменной Y присваивается столбец «Состояние каната»).

5. Построение и обучение нейронной сети посредством написания следующего кода (данный программный код, состоящий из двух строк, заменяет написание программного кода нейронной сети «вручную» из предыдущего алгоритма):

nn=MLPClassifier(hidden_layer_sizes=(8,8,8), max_iter=2000) (задается архитектура нейронной сети, а также параметры ее обучения; в данном случае 3 скрытых слоя, в каждом по 8 нейронов; количество эпох обучения 2000; по умолчанию выбраны функция активации Relu (код: activation='relu'), функция ошибки Adam (код: solver='adam'), скорость обучения 0,0001 (код: alpha=0.0001). Имеется возможность выбора различных параметров обучения нейронной сети через написание соответствующих программных кодов через запятую. Со всевозможными функциями активации, функциями ошибки и другими параметрами сети, их кодами можно ознакомиться на официальном сайте интерактивной библиотеки Scikit-learn;

nn.fit(X, y) (внедрение в нейронную сеть ранее заданных параметров X — входные переменные (браковочные показатели стальных канатов) и Y — выходная целевая переменная (техническое состояние стального каната).

После запуска данного блока нейронная сеть построена по заданной архитектуре и обучена по заданным параметрам обучения на обучающих данных (выборках), загруженных через таблицу файла Excel. Условие остановки обучения — достижение контрольной производительности сети в 95 %. Процесс обучения занял около 1,5 минуты.

Далее приведены процедуры сохранения и использования ИНС с применением интерактивной библиотеки Scikit-learn:

1. Сохранение обученной нейронной сети в отдельный файл написанием следующего программного кода:

joblib_file = "joblib_model.pkl" (создание файла joblib_model.pkl);

joblib.dump(nn, joblib_file) (сохранение нейронной сети nn в файле joblib_model.pkl).

После запуска данного блока программа создаст файл joblib_model.pkl в той же папке, в которой расположен сам файл программы.

2. Создание нового файла в PyCharm с разрешением py или в Jupyter с разрешением ipynb (файл в Jupyter создается для удобства, далее файл так или иначе должен будет сохранен в формате py).

3. Импорт (в новый файл) интерактивных библиотек написанием следующего программного кода:

import joblib (библиотека для сохранения и внедрения отдельных фрагментов программного кода);

import PySimpleGUI as sg (библиотека для графического оформления программы).

4. Загрузка в новый файл обученной нейронной сети через файл joblib_model.pkl, написание следующего программного кода:

joblib_file = "D:\Python\PycharmProjects/joblib_model.pkl" (указание полного пути к файлу joblib_model.pkl (индивидуально для каждого пользователя).

joblib_nn = joblib.load(joblib_file) (загрузка программного кода с обученной нейронной сетью).

5. Разработка графического оформления программы написанием следующего программного кода:

```
sg.theme('DarkAmber')
layout = [ [sg.Text('Определение состояния стального каната')];
[sg.Text('Введите 10 чисел через пробел'), sg.InputText()];
[sg.Button('Ok'), sg.Button('Cancel')]
window = sg.Window('Window Title', layout)
while True:
event, values = window.read()
if event == sg.WIN_CLOSED or event == 'Cancel':
break
#Нейронная сеть
Xnew = [list(map(int, values[0].split()))]
y = joblib_nn.predict(Xnew)
#Всплывающее окно
if event == 'Ok':
window.disappear()
sg.popup('Состояние каната ', y)
window.reappear()
print('Состояние каната ', y)
window.close()
```

После запуска данного блока программа выводит следующее рабочее окно (рис. 3).

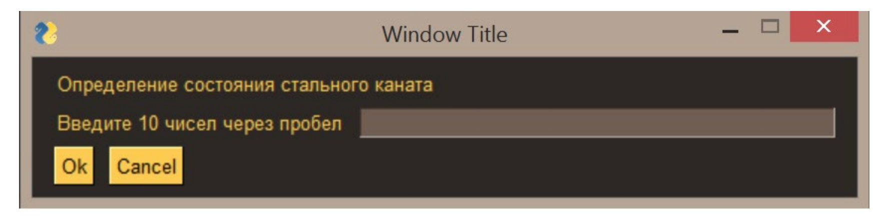


Рис. 3. Рабочее окно написанной программы

В рабочем окне программы имеется строка для ввода 10 значений через пробел. В данной строке вводятся фактические проценты десяти различных дефектов стальных канатов от их допустимых значений. После ввода, например, следующих значений [20 0 30 0 0 10 0 0 0 0] и нажатия клавиши «Ок» программа переводит пользователя в следующее рабочее окно.

В рабочем окне (рис. 4) программа выводит информационное сообщение «Состояние каната [1]», что соответствует состоянию каната «работоспособное, эксплуатация разрешена».

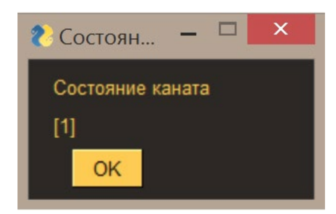


Рис. 4. Рабочее окно программы с определенным состоянием каната

Результаты исследования. Таким образом, разработаны две ИНС, имеющие одинаковую архитектуру, параметры обучения, состав, количество и объем обучающих, контрольных и тестовых выборок, но построенные разными методами. Результаты работы ИНС, построенной в программном комплексе STATISTICA, представлены в таблице 2.

https:// bps-journal.ru

Таблица 2

Результаты работы ИНС, построенной в STATISTICA

Выборка (тест)	Целевая	Выход сети	Состояние каната – 1 (доверительный уровень)	Состояние каната – 2 (доверительный уровень)	Состояние каната – 3 (доверительный уровень)
1	1	1	0,64	0,18	0,18
2	2	2	0,15	0,73	0,12
3	2	2	0,24	0,69	0,15
4	3	3	0,03	0,28	0,69
5	3	3	0,15	0,21	0,64
6	1	1	0,55	0,31	0,14
7	3	3	0,03	0,21	0,76
8	1	1	0,68	0,21	0,11
9	2	2	0,26	0,61	0,13
10	2	2	0,14	0,53	0,33

В столбце «Состояние каната — Целевая» показано заранее известное состояние каната, в столбце «Состояние каната — Выход» — результат работы нейронной сети; последующие столбцы указывают доверительные вероятности определения того или иного состояния каната, которые соответствуют уровням активации выходных нейронов сети.

По данным таблицы 2 видно, что нейронная сеть правильно определила состояние стального каната по браковочным показателям в 10 из 10 случаев, т. е. тестовая производительность составила 100 %. При этом среднее значение доверительных уровней определения состояния канатов составило 0,65.

Результаты работы ИНС, построенной с использованием интерактивной библиотеки Scikit-learn, представлены в таблице 3.

Таблица 3

Результаты работы ИНС, построенной с использованием Scikit-learn

Выборка (тест)	Целевая	Выход сети	Состояние каната – 1 (доверительный уровень)	Состояние каната – 2 (доверительный уровень)	Состояние каната – 3 (доверительный уровень)
1	1	1	0,58	0,13	0,29
2	2	2	0,19	0,51	0,19
3	2	2	0,22	0,56	0,30
4	3	3	0,29	0,25	0,46
5	3	3	0,21	0,22	0,57
6	1	1	0,57	0,22	0,21
7	3	3	0,21	0,22	0,57
8	1	1	0,63	0,22	0,15
9	2	2	0,22	0,57	0,21
10	2	2	0,33	0,45	0,21

По данным таблицы 3 видно, что нейронная сеть правильно определила состояние стального каната по браковочным показателям в 10 из 10 случаев, т. е. тестовая производительность составила 100 %. При этом среднее значение доверительных уровней определения состояния канатов составило 0,55.

Сравнения тестовой производительности нейронных сетей с одинаковой архитектурой и параметрами обучения, но построенных разными методами, показало, что ИНС, построенная на базе программного комплекса STATISTICA, и ИНС, построенная с использованием интерактивной библиотеки Scikit-learn, имеют тестовую производительность в 100 % при объеме тестовой выборки 10. Однако средний доверительный уровень (уровень активации «победившего» нейрона) определения состояния стального каната ИНС, построенной на базе программного комплекса STATISTICA, составил 0,65, тогда как тот же показатель ИНС, построенной с использованием интерактивной библиотеки Scikit-learn, составил 0,55, что на 15 % меньше.

Обсуждение и заключение. Полученные результаты показали, что ИНС, построенная с использованием программного комплекса STATISTICA, при одинаковой архитектуре и параметрах обучения сети имеет более оптимальные программные алгоритмы по критериям доверительной вероятности оценки технического состояния стального каната и скорости обучения сети, в сравнении с ИНС, построенной с использованием бесплатной

библиотеки Scikit-learn. Это можно объяснить тем, что при разработке алгоритмов программных комплексов типа STATISTICA используются специализированные аппаратные комплексы, включающие в себя векторные и тензорные процессоры, которые выходят далеко за рамки возможностей среднестатистического прикладного разработчика и требуют привлечения высококвалифицированных специалистов. Однако показатель тестовой производительности ИНС оказался одинаковым для обеих ИНС. Важно отметить, что при оценке данного показателя объем тестовой выборки составил 10. При увеличении тестовой выборки данный показатель будет точнее. При всем этом достигнутый результат обосновывает использование ведущими мировыми научно-исследовательскими и коммерческими центрами в области искусственного интеллекта библиотек TensorFlow, PyTorch, Scikit-learn.

Кроме того, полученный научный вывод позволит численно оценить и сравнить качество ИНС, имеющих одинаковую архитектуру и параметры обучения, но построенных различными методами, и может быть полезен как для будущих научных исследований в этой области, так и для выбора оптимальной среды построения ИНС в промышленной сфере. Разработанные программы могут применяться специалистами и экспертами как интеллектуальные системы поддержки принятия решений при диагностировании технического состояния стальных канатов.

Список литературы / References

1. Жернаков С.В. Применение технологии нейронных сетей для диагностики технического состояния авиационных двигателей. *Интеллектуальные системы в производстве*. 2006;2(8):70–83.

Zhernakov SV. Application of Neural Network Technology to Diagnose the Technical Condition of Aircraft Engines. *Intellektual'nye Sistemy v Proizvodstve*. 2006;2(8):70–83. (In Russ.).

2. Панфилов А.В., Николаев Н.Н., Хван Р.В. Короткий А.А. Оценка возможных аварий канатных дорог по компетенциям работников с использованием нейронных сетей. *Научно-технический вестник Брянского государственного университета*. 2023;(1):79–86. https://doi.org/10.22281/2413-9920-2023-09-01-79-86

Panfilov AV, Nikolaev NN, Khvan RV, Korotkiy AA. Assessment of Possible Cable Car Accidents by Employee Competencies Using Neural Networks. *Nauchno-Tekhnicheskiy Vestnik Bryanskogo Gosudarstvennogo Universiteta*. 2023;(1):79–86. https://doi.org/10.22281/2413-9920-2023-09-01-79-86 (In Russ.).

3. Горева Т.И., Портнягин Н.Н., Пюкке Г.А. Нейросетевые модели диагностики технических систем. *Вестник КРАУНЦ. Физ.-мат. науки*. 2012;1(4):31–43.

Goreva TI, Pornjagin NN, Pjukke GA. Neural Network Model Diagnosis Technical Systems. *Bulletin of the Kamchatka Regional Association Educational and Scientific Center (KRASEC). Physicsal and Mathematicsal Sciences.* 2012;1(4):31–43. (In Russ.).

4. Beskopylny AN, Shcherban EM, Stelmakh SA, Mailyan LR, Meskhi B, Razveeva I, et al. Discovery and Classification of Defects on Facing Brick Specimens Using a Convolutional Neural Network. *Applied Sciences.* 2023;13(9):5413. https://doi.org/10.3390/app13095413

5. Stelmakh SA, Shcherban EM, Beskopylny AN, Mailyan LR, Meskhi B, Razveeva I, et al. Prediction of Mechanical Properties of Highly Functional Lightweight Fiber-Reinforced Concrete Based on Deep Neural Network and Ensemble Regression Trees Methods. *Materials.* 2022;15(19):6740. https://doi.org/10.3390/ma15196740

6. Beskopylny AN, Stelmakh SA, Shcherban EM, Mailyan LR, Meskhi B, Razveeva I, et al. Concrete Strength Prediction Using Machine Learning Methods CatBoost, k-Nearest Neighbors, Support Vector Regression. *Applied Sciences.* 2022;12(21):10864. https://doi.org/10.3390/app122110864

7. Beskopylny AN, Shcherban EM, Stelmakh SA, Mailyan LR, Meskhi B, Razveeva I, et al. Detecting Cracks in Aerated Concrete Samples Using a Convolutional Neural Network. *Applied Sciences.* 2023;13(3):1904. https://doi.org/10.3390/app13031904

8. Воронцов В.А., Федоров Е.А. Разработка прототипа интеллектуальной системы оперативного мониторинга и технического состояния основных бортовых систем космического аппарата. *Труды МАИ*. 2015;2:1–35.

Vorontsov VA, Fedorov EA. Development of a Prototype of an Intelligent System for Operational Monitoring and Technical Condition of the Main Onboard Systems of the Spacecraft. *Trudy MAI*. 2015;2:1–35. (In Russ.).

9. Панфилов А.В., Месхи Б.Ч., Короткий А.А., Юсупов А.Р., Хван Р.В. *Программно-аппаратный комплекс визуально-измерительного контроля стальных канатов на основе компьютерного зрения и искусственного интеллекта.* Монография. Ростов-на-Дону: Донской государственный технический университет; 2023. 131 с.

Panfilov AV, Meskhi BCh, Korotkiy AA, Yusupov AR, Khvan RV. *Software and Hardware Complex for Visual and Measuring Control of Steel Ropes Based on Computer Vision and Artificial Intelligence.* Monograph. Rostov-on-Don: DSTU; 2023. 131 p. (In Russ.).

10. Панфилов А.В., Николаев Н.Н., Юсупов А.Р., Короткий А.А. Интегральная оценка риска при диагностике стальных канатов с использованием компьютерного зрения. *Безопасность техногенных и природных систем*. 2023;(1):56–69. https://doi.org/10.23947/2541-9129-2023-1-56-69

Panfilov AV, Nikolaev NN, Yusupov AR, Korotkiy AA. Integral Risk Assessment in Steel Ropes Diagnostics Using Computer Vision. *Safety of Technogenic and Natural Systems.* 2023;(1):56–69. https://doi.org/10.23947/2541-9129-2023-1-56-69

11. Seyed Reza Ghoreishi, Tanguy Messager, Cartraud P, Davies P. Validity and Limitations of Linear Analytical Models for Steel Wire Strands under Axial Loading, Using a 3D FE Model. *International Journal of Mechanical Sciences.* 2007;49(11):1251–1261. https://doi.org/10.1016/j.ijmecsci.2007.03.014

12. Frikha A, Cartraud P, Treyssède F. Mechanical Modeling of Helical Structures Accounting for Translational Invariance. Part 1: Static Behavior. *International Journal of Solids and Structures.* 2013;50(9):1373–1382. https://doi.org/10.1016/j.ijsolstr.2013.01.010

13. Короткий А.А., Панфилов А.В., Хван Р.В., Юсупов А.Р. Интегральный метод оценки дефектов на работоспособность стального каната с использованием искусственных нейронных сетей. *Транспортное, горное и строительное машиностроение: наука и производство*. 2023;8:73–79. https://doi.org/10.26160/2658-3305-2023-18-73-79

Korotkiy AA, Panfilov AV, Khvan RV, Yusupov AR. Integral Method of Assessing Defects on the Operability of Steel Rope Using Artificial Neural Networks. *Transport, mining and construction engineering: science and production.* 2023;8:73–79. https://doi.org/10.26160/2658-3305-2023-18-73-79 (In Russ.).

14. Foti F, De Luca di Roseto A. Analytical and Finite Element Modelling of the Elastic–Plastic Behaviour of Metallic Strands under Axial–Torsional Loads. *International Journal of Mechanical Sciences.* 2016;115–116:202–214. https://doi.org/10.1016/j. ijmecsci.2016.06.016

15. Spak K, Agnes G, Inman D. Cable Modelling and Internal Damping Developments. *Applied Mechanics Reviews.* 2013;65(1):010801. https://doi.org/10.1115/1.4023489

Об авторе:

Роман Владимирович Хван, кандидат технических наук, старший преподаватель кафедры эксплуатации транспортных систем и логистики Донского государственного технического университета (344003, РФ, г. Ростов-на-Дону, пл. Гагарина, 1), SPIN-код: 8662-6094, ORCID, ResearcherID, ScopusID, khvanroman@yandex.ru

Конфликт интересов: автор заявляет об отсутствии конфликта интересов.

Автор прочитал и одобрил окончательный вариант рукописи.

Поступила в редакцию 12.03.2024
Поступила после рецензирования 05.04.2024
Принята к публикации 11.04.2024

About the Author:

Roman V. Khvan, Cand.Sci. (Eng.), Senior Lecturer of the Operation of Transport Systems and Logistics Department, Don State Technical University (1, Gagarin Sq., Rostov-on-Don, 344003, RF), SPIN-code: 8662-6094, ORCID, ResearcherID, ScopusID, khvanroman@yandex.ru

Conflict of Interest Statement: the author does not have any conflict of interest.

The author has read and approved the final version of the manuscript.

Received 12.03.2024
Revised 05.04.2024
Accepted 11.04.2024